

100202410-1

United States Patent Application

For

ROUND ROBIN ARBITRATION SYSTEM

Inventor:

Bruce E. LaVigne

## ROUND ROBIN ARBITRATION SYSTEM

### BACKGROUND OF THE INVENTION

#### 5 FIELD OF THE INVENTION

The present invention generally relates to arbitration systems. More particularly, the present invention relates to round robin arbitration systems.

### RELATED ART

10 A device or a system has numerous resources. One or more requesters may request to use the numerous resources. Round robin arbitration ensures that the requesters have equal access to the numerous resources. A request bit corresponds to each requester.

15 Round robin arbitration is a manner of choosing all elements in a list, set, or group equally in some rational order, usually from the top to the bottom of a list and then starting again at the top of the list and so on. In addition, round robin arbitration is characterized by sequential, cyclical allocation of resources to numerous requesters.

20 Generally, a selection in a round robin arbitration is based on the previous selected request bit and the current request bits seeking access to a particular resource. Moreover, the round robin arbitration outputs an index corresponding to the selected request bit.

25 In conventional round robin arbiters, the entire round robin arbitration is typically performed with combinational logic in one clock cycle. However, as the clock frequency increases, it becomes difficult to design large bit width round robin arbiters using only combinational logic.

30

In an effort to overcome the difficulties associated with large scale combinational logic, some designers have divided their round robin arbiter into multiple stages. In effect, one stage chooses the sub-block which the next stage uses. The second stage tracks the last selection from each sub-block. Unfortunately, this uses many flip-flops and has the added disadvantage that it does not produce results in true round-robin fashion.

In yet another conventional round robin arbiter design approach, true round robin arbitration is performed. However, all variables for each arbitration are re-computed. Thus, in this case arbitration results are produced only once every 2 clock cycles.

Therefore, the typical prior art round robin arbitration schemes are all problematic and suffer different drawbacks.

## SUMMARY OF THE INVENTION

- A round robin arbitration system is disclosed. The round robin arbitration system includes a first round robin arbitration module and a second round robin arbitration module. The first round robin arbitration module has a first bit width. Moreover, it is configured to partition a plurality of requests into a plurality of blocks of requests, to select a block having one or more active requests using round robin arbitration, and to generate a first index corresponding to the selected block.
- 10        The second round robin arbitration module has a second bit width. Moreover, it is configured to store each request of the selected block, to select each active request of the selected block using round robin arbitration, to generate a second index corresponding to the selected active request, and to generate a first signal for synchronizing operation of the first and second round robin modules. In addition, the round robin arbitration system has a bit width that is a product of the first and second bit widths.
- 15

## BRIEF DESCRIPTION OF THE DRAWINGS

The accompanying drawings, which are incorporated in and form a part of this specification, illustrate embodiments of the invention and, together with the description, serve to explain the principles of the present  
5 invention.

Figure 1 illustrates a block diagram of a round robin arbitration system in accordance with an embodiment of the present invention.

10 Figure 2 illustrates a circuit diagram of a round robin arbitration system in accordance with an embodiment of the present invention.

15 Figures 3A-3C illustrate how requests are processed by a round robin arbitration system in accordance with an embodiment of the present invention.

Figure 4 illustrates a flow chart showing a method of arbitrating among a plurality of requests in accordance with an embodiment of the present invention.

20 The drawings referred to in this description should not be understood as being drawn to scale except if specifically noted.

## DETAILED DESCRIPTION OF THE INVENTION

Reference will now be made in detail to the preferred embodiments of the present invention, examples of which are illustrated in the accompanying drawings. While the invention will be described in conjunction with the 5 preferred embodiments, it will be understood that they are not intended to limit the invention to these embodiments. On the contrary, the invention is intended to cover alternatives, modifications and equivalents, which may be included within the spirit and scope of the invention as defined by the appended claims. Furthermore, in the following detailed description of the 10 present invention, numerous specific details are set forth in order to provide a thorough understanding of the present invention.

According to an embodiment of the present invention, a round robin arbitration system uses a two-stage design. Moreover, one of the stages 15 generates a signal to synchronize the operation of the first stage and the second stage. Unlike prior arbitration system designs, the round robin arbitration system can have large bit widths and still operate at high clock frequencies.

20       Figure 1 illustrates a block diagram of a round robin arbitration system 100 in accordance with an embodiment of the present invention. The round robin arbitration system 100 includes a first stage round robin arbitration module 10 and a second stage round robin arbitration module 20. The round robin arbitration system 100 is able to operate at higher clock 25 frequencies than the conventional single stage round robin arbiter. Moreover, the round robin arbitration system 100 produces true round robin results, one result every clock cycle. Although two stages were used in this embodiment, the round robin arbitration system 100 can have more than two stages. The additional stages widen the bit width of the round robin 30 arbitration system 100 such that the bit width of the round robin arbitration system 100 becomes the product of each bit width of each stage.

Specifically, the round robin arbitration system 100 can arbitrate among large numbers of requests at high clock frequencies. As illustrates in Figure 1, the round robin arbitration system 100 receives N requests as 5 input. In an embodiment, N is 256. It should be understood that N can be any other number.

The N requests are inputted into the first stage round robin arbitration module 10 and the second stage round robin arbitration module 20. The 10 first stage round robin arbitration module 10 outputs a first index (i.e., UPPER) which is A bits for the selection of the request by the round robin arbitration system 100. The second stage round robin arbitration module 20 outputs a second index (i.e., LOWER) which is B bits for the selection of the request by the round robin arbitration system 100. The combination of the 15 first index and the second index forms the index corresponding to the selection of the request by the round robin arbitration system 100. For example, if N is 256, the first index is four bits and the second index is four bits. Thus, the 8-bit index corresponds to the request that has been selected by the round robin arbitration system 100 from the plurality of received 20 requests.

As illustrated in Figure 1, the second stage round robin arbitration module 20 outputs a signal 30 for synchronizing the operation of the first stage round robin arbitration module 10 and the second stage round robin arbitration module 20. Moreover, a valid signal 40 is generated based on information from the first stage round robin arbitration module 10 and the second stage round robin arbitration module 20. The valid signal 40 indicates whether the first index (i.e., UPPER) and the second index (i.e., LOWER) are valid.

Figure 2 illustrates a circuit diagram of a round robin arbitration system 100 in accordance with an embodiment of the present invention. As illustrated in Figure 2, the round robin arbitration system 100 has a 256 bit width. Thus, the round robin arbitration system 100 can arbitrate among 256 requests. The round robin arbitration system 100 includes a first stage 10 and a second stage 20. It should be understood that the round robin arbitration system 100 can be configured to other bit widths.

The first stage 10 receives the 256 request bits, each corresponding to a requester. A request is active ("High") if the corresponding requester wants to access a resource. Otherwise, the request is inactive ("low"). Figure 3A illustrates the 256 request bits  $R_{256} - R_0$  that are received by the first stage 10. Continuing with Figure 2, in the first stage 10, the 256 requests are coupled to a reduction-OR circuit 210. The reduction-OR circuit 210 partitions the 256 request bits into 16 blocks of 16 request bits each. Figure 3B illustrates how the request bits  $R_{256} - R_0$  are partitioned into blocks (e.g., Block15 - Block0). Moreover, the reduction-OR circuit 210 performs an OR operation on each block of request bits to generate a block bit (e.g.,  $B_{15} - B_0$ ) for each of the 16 blocks, as illustrated in Figure 3C. The block bit indicates whether the corresponding block has one or more active request bits.

Continuing with Figure 2, the output (i.e., 16 bits) of the reduction-OR circuit 210 is inputted into the Arbiter1, which is a 16-bit round robin arbiter. The Arbiter1 selects a block bit corresponding to a block having one or more active request bits based on round robin arbitration and the previous selection *prev\_sel*. Moreover, the Arbiter1 generates a new selection index *new\_sel* corresponding to the selected block bit. For example, if block1 is selected, the new selection index *new\_sel* is 0001, a four bit index.

- The new selection index new\_sel (or block\_result) is inputted into the multiplexor MUX1. The selector of MUX1 is controlled by an OR gate 220. The OR gate 220 has a first inverted input coupled to the Found terminal of the Arbiter1 and a second input coupled to the another signal 230 which is
- 5 outputted by the Arbiter2, whereas the another signal 230 synchronizes the first stage 10 and the second stage 20. The Found terminal of the Arbiter1 indicates a valid arbitration result by the Arbiter1. Moreover, the output of MUX1 is coupled to flip-flop FF1. In addition, the output of FF1 is coupled to MUX2 and is inputted into MUX1 and into the previous selection prev\_sel
- 10 terminal of the Arbiter1. The selector of MUX2 is controlled by another signal 230. The output of MUX2 is coupled to FF2. The output of FF2 is inputted into MUX2. Moreover, FF2 outputs the UPPER bits (i.e., four bits) of the selected request bit made by the round robin arbitration system 100.
- 15 The second stage 20 receives the 256 request bits, each corresponding to a requester. The 256 request bits are inputted into the multiplexor MUX3. The selector of MUX3 is controlled by the output of FF1, whereas the output of FF1 indicates the selected block. The MUX3 outputs the request bits (i.e., 16 request bits) of the selected block and couples the
- 20 request bits (i.e., 16 request bits) to MUX4 and to MUX5. MUX4 is coupled to FF3. The selector of MUX4 is controlled by another signal 230. The output of FF3 is coupled to MUX5 and MUX4. The selector of MUX5 is controlled by another signal 230. Moreover, the output of MUX5 is inputted into Arbiter2 to enable the Arbiter2 to arbitrate among the active request bits
- 25 of the selected block. FF3 stores the request bits of the selected block so that Arbiter2, which is a 16-bit round robin arbiter, can arbitrate among the active request bits of the selected block.
- 30 The Arbiter2 selects each of the active request bits of the selected block based on round robin arbitration and the previous selection prev\_sel. Moreover, the Arbiter2 generates a new selection index new\_sel.

corresponding to the selected active request bit. For example, if active request bit 4 is selected, the new selection index new\_sel is 0100, a four bit index. The new selection index new\_sel terminal of Arbiter2 is coupled to FF5. Moreover, FF5 outputs the LOWER bits (i.e., four bits) of the selected 5 request bit made by the round robin arbitration system 100. The Arbiter2 outputs (via terminal MORE) a signal which is coupled to the FF4 which outputs another signal 230 which synchronizes the first stage 10 and the second stage 20. The signal at the terminal MORE goes "high" to indicate that the Arbiter2 has produced a first arbitration result by selecting one of the 10 active request bits of the selected block and to indicate that there is another active request bit available (in the selected block) for selection before the Arbiter2 wraps. As illustrated in Figure 2, the output of MUX6 is coupled to the previous selection prev\_sel terminal of Arbiter2. The selector of MUX6 is controlled by another signal 230. Moreover, the output of FF5 is inputted 15 into MUX6. The wrap value 15 (or 1111) is also inputted into MUX6, whereas the wrap value 15 (or 1111) causes the Arbiter2 to wrap.

In addition, the VALID signal is outputted by the FF6, whereas the valid signal indicates whether the UPPER index generated by the first stage 20 10 is valid and whether the LOWER index generated by the second stage 20 is valid. The VALID signal prevents the same active request bit from being selected two clocks cycles in a row. In particular, the output (i.e., next\_UPPER) of MUX2 and the output (i.e., UPPER) of FF2 are compared 250 to determine whether they are not equal. The output (i.e., next\_LOWER) of new\_sel terminal of Arbiter2 and the output (i.e., LOWER) of FF5 are compared 260 to determine whether they are not equal. The result of compare 250, the result of compare 260, and the Found terminal of Arbiter2 (whereas the Found terminal of the Arbiter1 indicates a valid arbitration 270 result by the Arbiter2) are inputted into the AND gate 270. The output of the 30 AND gate 270 is coupled to the FF6 which outputs the VALID signal.

It should be understood that Arbiter1 and Arbiter2 can have bit widths other than 16 bits. The bit width of the Arbiter1 and the bit width of the Arbiter2 depend on desired bit width for the round robin arbitration system 100. The round robin arbitration system 100 has a bit width that is a product 5 of the bit width of the Arbiter1 and the bit width of the Arbiter2.

In practice, after Arbiter1 (of the first stage 10) selects a block based on round robin arbitration, the Arbiter2 (of the second stage 20) selects an active request bit from the selected block. If the MORE terminal goes "High", 10 another signal 230 also goes "High" after one clock cycle. Although Arbiter1 can proceed to select the next block, another signal 230 causes MUX1 and MUX2 to maintain the UPPER index of the selected block, which is being processed by the Arbiter2, rather than changing the UPPER index to correspond with the next selected block.

15

Similarly, once another signal 230 goes "High", MUX4 and MUX5 continue to output the request bits of the selected block rather than the request bits of the next selected block. The FF3 stores the request bits of the selected block. Hence, the Arbiter2 can proceed processing the active 20 request bits of the selected block. Moreover, another signal 230 causes the Arbiter2 to receive (via MUX6) at its prev\_sel terminal the output of the FF5 to perform round robin arbitration rather than causing the Arbiter2 to receive the wrap value 15 (or 1111) which causes the Arbiter2 to wrap. In sum, another signal synchronizes the operation of the first phase 10, the second 25 phase 20, Arbiter1, and Arbiter2.

Figure 4 illustrates a flow chart showing a method 400 of arbitrating among a plurality of requests in accordance with an embodiment of the present invention.

30

At Block 410, the requests are partitioned into a plurality of blocks of requests. At Block 420, a particular block having one or more active requests using a first round robin arbiter. At Block 430, each request of the particular block is stored.

5

Continuing at Block 440, each active request of the particular block is selected using a second round robin arbiter. At Block 450, the selection of the particular block and the selection of the active request of the particular block are synchronized.

10

Therefore, a round robin arbitration system has been described. The foregoing descriptions of specific embodiments of the present invention have been presented for purposes of illustration and description. They are not intended to be exhaustive or to limit the invention to the precise forms disclosed, and many modifications and variations are possible in light of the above teaching. The embodiments were chosen and described in order to best explain the principles of the invention and its practical application, to thereby enable others skilled in the art to best utilize the invention and various embodiments with various modifications as are suited to the particular use contemplated. It is intended that the scope of the invention be defined by the Claims appended hereto and their equivalents.